**3: Event Ranking System (Heap Sort)**: Implement **heap sort** to rank participants in a large-scale competition based on their scores. Test your solution with large datasets.

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Event Ranking with Heap Sort</title>

<style>

body { font-family: Arial, sans-serif; padding: 20px; }

button { padding: 10px 15px; cursor: pointer; }

pre { margin-top: 20px; font-size: 14px; }

</style>

</head>

<body>

<h1>Event Ranking System</h1>

<button onclick="rankParticipants()">Rank Participants</button>

<pre id="rankingResult"></pre>

<script>

const participants = [

{ name: "John", score: 90 },

{ name: "Jane", score: 80 },

{ name: "Alex", score: 95 },

{ name: "Chris", score: 85 },

{ name: "Sarah", score: 92 },

{ name: "Max", score: 75 },

{ name: "Mike", score: 88 },

{ name: "Zoe", score: 91 }

];

function heapSort(arr) {

let n = arr.length;

for (let i = Math.floor(n / 2) - 1; i >= 0; i--) {

heapify(arr, n, i);

}

for (let i = n - 1; i > 0; i--) {

[arr[0], arr[i]] = [arr[i], arr[0]];

heapify(arr, i, 0);

}

return arr;

}

function heapify(arr, n, i) {

let largest = i;

let left = 2 \* i + 1;

let right = 2 \* i + 2;

if (left < n && arr[left].score > arr[largest].score) largest = left;

if (right < n && arr[right].score > arr[largest].score) largest = right;

if (largest !== i) {

[arr[i], arr[largest]] = [arr[largest], arr[i]];

heapify(arr, n, largest);

}

}

function rankParticipants() {

const start = performance.now();

const sortedParticipants = heapSort(participants);

const end = performance.now();

const result = sortedParticipants.map(p => `${p.name}: ${p.score}`).join("\n");

document.getElementById("rankingResult").textContent = `Ranked Participants:\n${result}\nExecution time: ${(end - start).toFixed(4)}ms`;

}

</script>

</body>

</html>